Exploration of architectural parameters for future HPC systems

Constantino Gomez*†, Francesc Martinez*†, Adria Armejach*†, Marc Casas*†, Filippo Mantovani*†, Miquel Moreto*
*Barcelona Supercomputing Center, Barcelona, Spain
†Universitat Politècnica de Catalunya, Barcelona, Spain
E-mail: {constantino.gomez, francesc.martinez, adria.armejach, marc.casas, filippo.mantovani, miquel.moreto}@bsc.es

Keywords—Co-Design, Large-scale simulations, High-performance computing.

I. Extended Abstract

Trends in High Performance Computing (HPC) systems are shifting. The use of commodity server processors as the main option to design these systems is moving towards a more specialized landscape. Processor trends are evolving in several directions, such as, leaner core designs [1], larger core counts per socket [2], wide vector units [3], or with integrated memory like high-bandwidth memory (HBM) modules via silicon interposer technologies [4].

In our work, we undertake a design space exploration study that considers the most relevant design trends we are observing today in HPC systems. To perform this study, we follow a recently introduced multi-level simulation methodology (MUSA) [5]. MUSA enables fast and accurate performance estimations and takes into account inter-node communication, node-level architecture, and system software interactions. Through our extensive design space exploration, we provide hardware and software co-design recommendations for next-generation large-scale HPC systems.

A. Co-Design opportunities

The design space for next-generation HPC machines is expanding. First, the trend to use commodity server processors as the common choice is changing towards processors with leaner core designs that feature different microarchitectural characteristics. For example, Cray has already deployed Isambard [6], a system with 10,000+ Armv8 cores; and now supports ARM-based processors (including the Cavium ThunderX2) across their main product line. Second, vector architectures with larger lengths than the ones employed in recent years are starting to be considered again. In this regard, Arm recently introduced the Scalable Vector Extensions (SVE) that support up to 2,048 bit vectors and per-lane predication. Third, several memory technologies are starting to appear in the HPC domain, for example: die-stacked DRAM like the one employed in Knights Landing [7], or High-Bandwidth Memory (HBM) already used in a number of GPUs.

The advent of these trends and technologies leads to a large design space for next-generation HPC machines that needs to be carefully considered. There is a clear opportunity to co-design hardware and software by mapping application requirements to the available hardware ecosystem that these trends are opening. In addition, the ability to predict and fine-tune application performance for selected hardware designs that are deemed of interest is of paramount importance to system architects.

L3:L2-caches	**Size / associativity / latency**	
Label	**L3**	**L2**
32M:256KB	32MB / 16 / 68	256kB / 8 / 9
64M:512KB	64MB / 16 / 70	512kB / 16 / 11
96M:1MB	96MB / 16 / 72	1MB / 16 / 13

Core OoO Label	**ROB**	**Issue& commit**	**Store buffer**	**#ALU/ #FPU**	**IRF/ FRF**
low-end	40	2	20	1 / 3	30/ 50
medium	180	4	100	3 / 3	130 / 70
high	224	6	120	4 / 3	180 / 100
aggressive	300	8	150	5 / 4	210 / 120

Other param.	**Values**
Frequency [GHz]	1.5, 2.0, 2.5, 3.0
Vector width [bits]	128, 256, 512
Memory [DDR4-2333]	4-channel, 8-channel
Number of Cores	1, 32, 64

TABLE I. Simulation architectural parameters and values used in our design space exploration including: cache size, associativity and latency; and OoO details like Reorder buffer (ROB) and Integer/Float Register File (RF).

B. Parameter exploration

After reviewing the HPC systems landscape, we select a set of important compute node features in current and upcoming HPC architectures. These features expose relevant energy and performance trade-offs when considering different HPC workloads. We focus our exploration on six features: number of cores in a socket, out-of-order (OoO) capabilities of the core, memory technology, floating-point unit (FPU) vector width, CPU frequency and cache size. Additionally, to do our simulations we select five relevant hybrid (MPI+OpenMP) HPC applications: HYDRO, SP-MZ, BT-MZ, SPECFEM3D and LULESH.

Table I shows a detailed list of all the parameters and values we explore and the names (labels) we will use to refer to them.

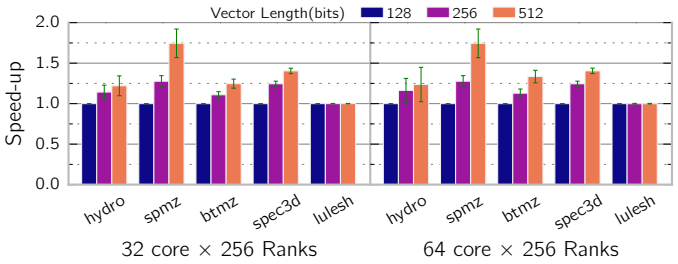

Fig. 1. Average performance speedup increasing FPU width up to 512-bits. Normalized to 128-bit configurations.

C. Results

Figure 1 summarizes the performance-energy trade-off when we increase the vector Floating Point (FP) registers used for SIMD operations in each core. Results for 32 and 64 core configurations are very similar. Excluding LULESH, wider 512-bit FP units yield 20% (HYDRO) to 75% (SP-MZ) application performance speed-up; 40% on average.

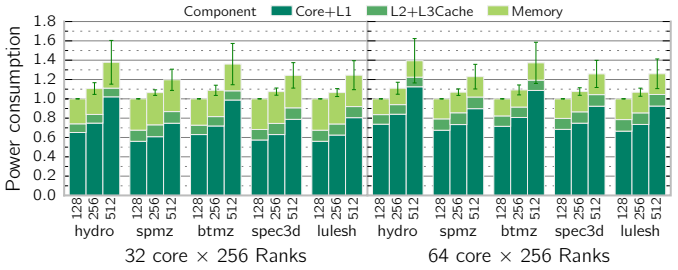

Fig. 2. Average power consumption increasing FPU width up to 512-bits. Normalized to 128-bit configurations.

In Figure 2, we see that using 512-bit vector width translates into an average power increment across applications of 60% with respect to 128-bit units in each core. As expected, the core power consumption is relatively larger in compute-intensive applications like HYDRO and BTMZ than in memory bound counterparts.

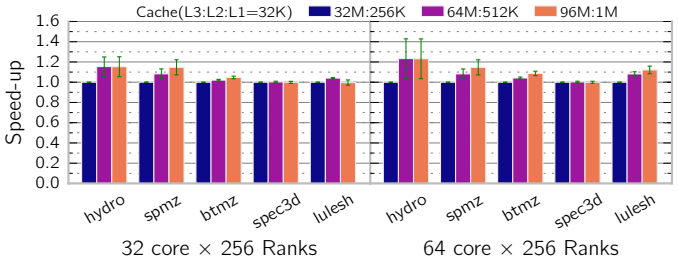

Fig. 3. Average performance speedup varying L3- and L2-cache parameters. Normalized to 32MB:256KB cache configs.

Figure 3 shows how only modifying L2- and L3-cache sizes affects performance in our simulations; at 64 cores, upgrading to a cache configuration with 96MB:1MB (1.5MB:1MB per core) results in an 11% average speedup across applications.

Taking into account these observations, we simulate parallel executions of SPMZ considering architectures with increasing SIMD widths of 1024- (*Vector+* configuration) and

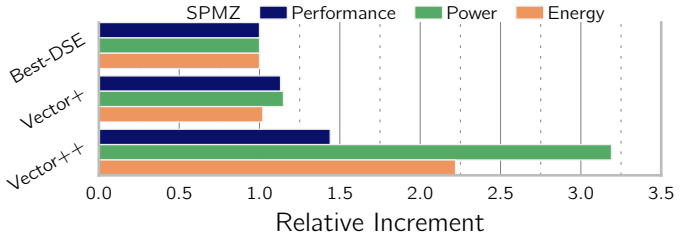

Fig. 4. Performance, power and energy-to-solution of a specific configurations targeting spmz.

2048-bits (*Vector++* configuration) while keeping the rest of the architectural features settings that give the best possible performance-power tradeoff.

D. Conclusion

In this study, we look at speedup and energy consumption exploring the design space (i.e., changing SIMD width, number of cores, and type of cores), and we provide architectural recommendations that can be used as hardware and software co-design guidelines targeting specific applications.

II. Acknowledgment

This work has been accepted as a conference paper and will be published in the proceedings of the International Parallel and Distributed Processing Symposium (IPDPS), 2019.

References

[1] F. Haohuan *et al.*, "The sunway taihulight supercomputer: system and applications," *SCIENCE CHINA Information Sciences*, vol. 59, no. 7, 2016.

[2] "Thunderx2 arm processors." [Online]. Available: cavium.com/product-thunderx2-arm-processors.html

[3] "SX-Aurora TSUBASA Architecture." [Online]. Available: nec.com/en/global/solutions/hpc/sx/architecture.html?

[4] "Fujitsu reveals details of processor that will power Post-K supercomputer." [Online]. Available: top500.org/news/fujitsu-reveals-details-of-processor-that-will-power-post-k-supercomputer/

[5] T. Grass *et al.*, "MUSA: a multi-level simulation approach for next-generation HPC machines," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC 2016*, pp. 526–537.

[6] "GW4 Isambard." [Online]. Available: gw4.ac.uk/isambard/

[7] A. Sodani, "Knights landing: 2nd generation intel xeon phi processor," in *2015 IEEE Hot Chips 27 Symposium (HCS)*, 2015, pp. 1–24.

Constantino Gómez is a third year Ph.D student at the Barcelona Supercomputing Center. He received the BSc and MSc degrees in Computer Science from the Universitat Politcnica de Catalunya (UPC) in 2014 and 2016. He has been involved as a researcher in the Mont-Blanc european project series since 2014. His research interests include simulation tools, emerging memory technologies and co-design for future massively parallel systems.